Stacking SRAM Banks for Ultra-Low Power Standby Mode Operation

Presenter: Adam C. Cabe, University of Virginia
Advisor: Mircea R. Stan, IEEE Senior Member, University of Virginia

***Abstract*—Minimizing leakage is essential for low power SRAM operation in the sub 90 nm regime. This work employs an implicit voltage reduction technique to SRAM, aimed to reduce leakage power expended during sleep mode. By stacking SRAM blocks, the voltage on each block is lowered close to the data retention voltage (DRV) of each cell - reducing leakage power by as much as 90% from the active power mode. Simulation results show the stability of the scheme around corners and process variations.**

I Introduction

The influx of portable media devices has sparked the need for novel low-power integrated-circuit design techniques aimed to increase product battery lifetime. Many of such techniques are applied towards SRAM, as these memory structures often occupy more than 50% of the die area, and consume excessive amounts of leakage power, even when idle. In recent years, power modes have proven effective in reducing the impact of both dynamic and leakage energy in such deep submicron technologies. Such techniques include reducing V_{dd} [1] [2], body-biasing [3], using multi-threshold CMOS [4] [5], and employing different SRAM cell structures [6] among many others.

Reducing V_{dd} is often employed as a power saving technique in SRAM banks. Scaling V_{dd} is typically achieved using a DC/DC converter, however overhead losses are accrued from the converters area and static power consumption. *Implicit* power conversion schemes [7] have recently been introduced for power delivery in order to reduce power supply currents. This work proposes to use a similar implicit power conversion scheme to reduce V_{dd} across an SRAM bank during standby mode, without the need for an explicit DC/DC converter.

The implicit power conversion technique works by "stacking" multiple SRAM banks in series, such that a voltage is divided between the blocks much like a resistive divider. This stacking is only applied during standby mode, so the SRAM banks operate at full V_{dd} while reading/writing during active mode. This allows for ultra-low voltage operation during standby, while still maintaining a high-performance level during active mode. Simulated results are presented through this work, using the 65nm PTM library to show performance at a current technology node. Results examine the overall power savings, the performance loss due to stacking multiple banks, any noise introduced as a result of the technique, and variation impacts on final design decisions.

II Stacking SRAM Banks to Reduce Leakage

The linear leakage power reduction with scaling provides large incentives to develop efficient methods to reduce voltage with minimum accrued overhead. DC/DC converters are typically used to achieve this reduced V_{dd} value. Common on-chip converters are the switch-capacitor (SC) and the buck converter. Several recent works have explored the design of efficient, ultra-low-power, and low-voltage DC/DC converters, and generally achieve these results in the efficiency range of 80 to 85 percent [2], [8].

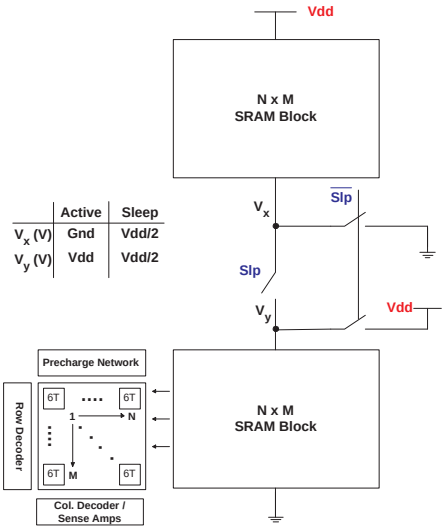

Fig. 1. Stacked SRAM where in active mode Slp = '0' and the SRAM is fully powered; in sleep mode Slp = '1' and the banks see half V_{dd}.

The intrinsic difficulty with DC/DC converters is they are designed to be efficient within a specified output load range. Furthermore, as this output load decreases, it becomes much more difficult to achieve such a high efficiency. This stems from the static control components, and dynamic switching components that consume some baseline energy regardless of the design topology. These low loads are becoming increasingly important particularly as designs scale to adapt to scavenged and harvested energy sources. The Phoenix processor, recently introduced in [9], employs SRAM memories as small as 260 bytes that are designed to operate near .5V. It is particularly difficult to build an efficient DC/DC converter in these output load ranges.

The stacked SRAM design in this work presents a simple way to achieve a reduced V_{dd} that does not require drastic design alterations based on SRAM size, and can provide high efficiencies at ultra-low loads. Fig. 8 shows a block diagram of the "stacked" SRAM employing the implicit power conversion technique. During active mode, Slp = '0' and the two SRAM blocks are supplied with full V_{dd}. When entering standby mode, the Slp signal goes high, and the ground node of the upper block is connected to the V_{dd} node of the lower block. The total V_{dd} is now shared between the upper and lower banks, and thus the voltage across each bank is "implicitly" reduced to $\frac{1}{2} \cdot V_{dd}$.

In practice, it is possible to stack more than two SRAM banks, and the stacking limitations are discussed throughout the remainder of this work. These limits largely stem from the data retention ability of the SRAM arrays. Furthermore, this design method must preserve the speed of the SRAM around process corners and variations. Therefore, the mechanisms used to implement the stacked banks need to be "transparent" during active mode, and robust to noise and on-chip variability. The next sec-

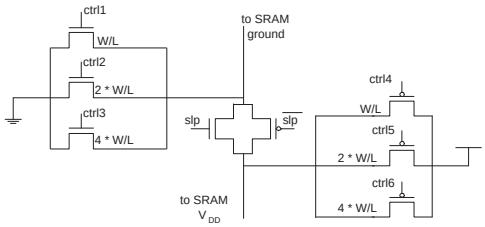

Fig. 2. Power switch consists of V_{dd} and ground header switches, and a transmission gate connecting the SRAM banks during sleep.

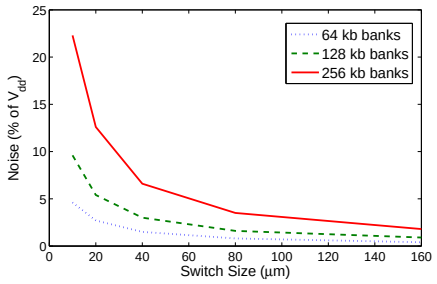

Fig. 3. Noise compared to power switch size. Size is shown per transistor, i.e. 80μm means both PMOS and NMOS headers are 80μm.

tions delve deeper into the stacking design concept, exploring the bank configurations, power switch requirements, and the impact of noise and variations on the final implementation.

III Stack Configuration and Power Switch Design

A *Power Switch Design*

The block diagram in Fig. 8 just shows a basic switch connecting the upper and lower SRAM banks during sleep. Fig. 2 elaborates on this switch, showing the detailed power switch design. The design is straightforward, consisting of a transmission gate connecting the two SRAM banks in sleep, and three transistors per rail to connect the SRAM to full supply during active mode. The reason three transistors are given per supply is to let the user adjust the amount of current supplied to the SRAM post-fabrication. The user can enable/disable signals ctrl1 through ctrl6 to allow more/less current to flow into the SRAM.

It is important to note that the power switches *only connect the bit cell arrays together* during sleep, not the peripheral circuitry such as the word-line drivers, decoders, pre-charge circuits, and sense amps. Only connecting the bit cell arrays together during sleep reduces the current delivery requirements on the power switches during active mode, as is elaborated on in the following sections.

The power switches control the current delivery to the cell array during both active and sleep modes. The size of the switches impacts the noise seen on these virtual supply rails, and the read speed of the SRAM. The write speed is largely unaffected since the write circuits forcibly overwrite the cell contents, which is driven from the write-driver power supply. The read speed can be impacted since the bit-cell must discharge the bit-line during a read. However fortunately during reads, only one row is activated at a time, largely reducing the load on the power switch. At the switch sizes examined in this work, simulation results generally show the speed reduction to be less than .5%.

Under dynamic activity, the virtual rails supplying current to the bit cells become noisy. There is a clear tradeoff between the size of the power switch and the noise seen on these virtual rails. Fig. 3 shows this tradeoff through simulation, comparing the area of the switches to the noise generated in various sizes of SRAM banks. This simulation is performed considering a two bank stack at the annotated sizes (i.e. each bank is 64kb). It is clear that for small banks, not much switch area is needed to keep noise to a minimum, however with increasing size, large switches are necessary to shunt the active mode current. Additional studies show that adding extra decoupling capacitance will also help decrease this noise, and may become necessary for very large banks [10].

The next section will briefly outline the ideas for power-down options for the decode circuitry. The focus on this work remains on the bit-cell leakage, however the ideas proposed will be studied into the future.

B *Reducing Power in Peripheral Circuits*

An SRAM is primarily comprised of the bit-cell array, precharge cells, row decoder, column decoder, and write/sense circuits. To this point, we have only considered stacking to reduce leakage within the bit-cells, however this section looks at power-down options for the peripheral circuits.

To begin, it is possible to stack the bit-cell array and all of the peripheral circuits with the same power switch shown in Fig. 2. This would reduce V_{dd} on every SRAM circuit during sleep, so what makes this option unattractive? A strong first reason is that stacking all circuits on the one power switch puts a heavy current load on the switch during active mode. This switch would now have to be much larger than those shown in Fig. 3 to keep the virtual rail noise to a minimum. Furthermore, if the designer chose to keep a smaller power switch, this increased noise is now directly present in the bit-cell array, which could cause stability issues to the SRAM cells.

There are several alternatives to stacking all circuits, however this work focuses on a subset of two choices. The first examines only stacking the bit cell arrays, leaving the surrounding circuits to operate at full V_{dd} during sleep. The second scheme is a hybrid scheme, where the bit cell array is stacked as in the first choice, however the decoder circuits are fully powered down using separate power header switches.

Stacking only the bit cell arrays presents the simplest solution where the peripheral circuits are left at full power at all times. This includes the row and column decoders, and the write circuits. Furthermore, this method eases the requirements on the power switches necessary to provide current to the bit cells. Since the decoders are powered on a separate supply, the power switches only need to be large enough to handle the leakage and small dynamic currents from the bit cells during read/write accesses.

The second method stacks the bit-cells and powers down the peripheral decode circuits, maximizing the power savings by completely turning the peripheral V_{dd} supply off during sleep. The peripheral circuits are powered down using separate header switches, and must be optimized for to maintain performance at the cost of switch area. Furthermore, keepers are added to the word-line driver outputs to ensure the word-lines are turned off during sleep, and do not float to an unknown value when entering or exiting sleep.

The next section begins to examine stacking more than two banks, and the limitations on the this scheme. These sections focus primarily on stacking only the bit-cell arrays, as mentioned in this section. Further optimizations for powering-down the peripheral circuits will be studied in the future, and similar studies have been conducted in the past [10], [11].

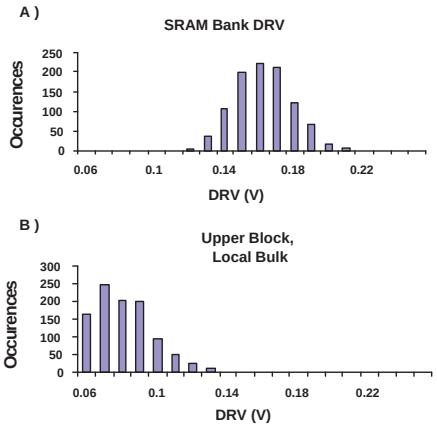

Fig. 4. **A.** DRV calculations for a single banks, assuming bodies are connected to virtual rails or left floating (SOI). **B.** DRV distribution of upper bank when connecting bulks to full-rail supply during sleep.

IV DRV Measurements and Limitations to Stacking

Thus far, this work focused on stacking two SRAM banks, and conceptually understanding how this technique works. In theory, stacking more than two banks should even further reduce power in sleep mode, however in practice there are limitations to stacking.

In order to safely enter sleep mode, we must ensure data retention when lowering the voltages across each SRAM bank. The data retention voltage (DRV) is the lowest rail-to-rail voltage at which a SRAM cell can retain its data, and is specific to the operating technology and process node of the design. Furthermore, this value heavily depends on SRAM design choices such as bit-cell sizing and type.

The DRV is calculated in this work to ensure that the stack scheme doesn't interfere with data retention. All of the DRV calculations shown here are done using a 90nm commercial technology to show the impact of more realistic variation data on SRAM stability then allowed by the PTM device models. The DRV of the SRAM cells is obtained by examining the static noise margin (SNM) of each cell [12].

Fig. 4A shows the DRV distribution plotted for this 90nm technology node considering a typical corner. The data was gathered using a monte carlo simulation run over 1,000 iterations. For this particular cell, the worst case DRV is near 200mV, which agrees with previous data from other authors [2].

The data in Fig. 4A is taken considering the SRAM bit-cell transistor bodies are connected to their local supply, or in other words, the virtual rail. By connecting the bulk nodes here, and having the bit-lines float during sleep, the leakage seen through each SRAM bank is essentially equal. Furthermore, this means the DRV for each bank is the same. This same result is achieved by leaving the bodies floating as seen in Silicon-on-Insulator (SOI) technologies.

Connecting the bit-cell bulk nodes to the full-rail power supply yields a leakage imbalance in the stack. This is illustrated in Fig. 5, which shows an SRAM cell in both the upper bank (A) and lower bank (B) in a two-bank stack. In the upper bank, the local ground is approximately $\frac{V_{dd}}{2}$. If the PMOS and NMOS bodies are connected to full rail supply, then the drive transistors (NMOS) become reverse body biased. This leads to a larger threshold voltage and smaller leakage through these drive transistors. A similar story happens in the lower bank, as detailed in fig. 5B, where the local V_{dd} supply is now approximately $\frac{V_{dd}}{2}$. If the bodies are connected to full rail supply, the pull-up devices become reverse body biased. The overall leakage imbalance stems from the body biasing having different strength impacts on the PMOS and NMOS devices. Even though the same body bias is applied in each stack, the effective resistance change is higher in the PMOS gates since these transistors have lower nominal threshold voltage magnitudes.

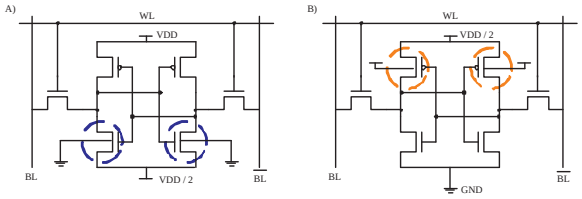

Fig. 5. **A.** 6T cell in the upper block of the stack. Circled NMOS gates are reverse body biased. **B.** Depicts a 6T cell in the lower block in the stack. Circled PMOS gates are reverse body biased.

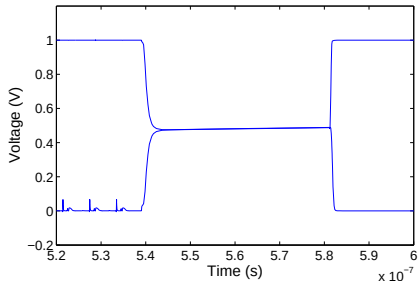

Fig. 6. Transient simulation of two stacked 4kb banks. Plot shows the virtual rails during active, and transitioning in and out of sleep.

This leakage imbalance will cause the intermediate voltage to drift away from $\frac{V_{dd}}{2}$ in the previous example, and will cause the DRV to shift at various points in the stack. This is shown in Fig. 4B, where the DRV in the upper bank decreases considerably over the lower bank. This shift only becomes a problem if the voltage drifts far enough to infringe on the banks DRV. This is not problematic for two-bank stacks, but becomes very prominent when stacking more than two banks.

Progressing forward in technology may also present challenges with SRAM data retention. This is shown in [13] where considering worst case process corners, the DRV can approach values up towards 400 mV in technology nodes beyond 90nm. This points out that one, it should be safe to stack at least two blocks during sleep mode, and that two, it is important to obtain information on the DRV of the memory style and working technology in use before employing this stacking technique. From the data in this work, and from this previous work [13], it appears that it could be safe to stack at least two banks. Stacking more than this will be dependent upon the DRV for the particular technology and working SRAM setup.

V Results

Fig. 6 shows a transient simulation result of the virtual power rails for two stacked 4kb SRAM banks. As discussed earlier, the noise on the power rails is present at less than 5% of V_{dd} while reading and writing the SRAM. The data outputs are not shown here, but the SRAM performs full swing read/write operations at approximately 350 MHz in the 65nm PTM technology.

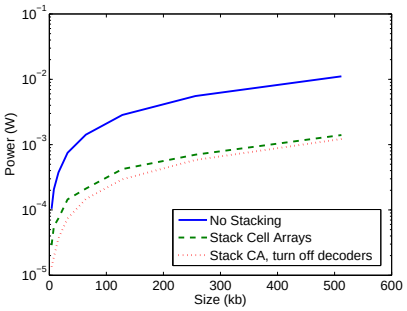

Fig. 7. Overall power savings of the two analyzed schemes. Shows leakage savings during sleep between $75\% - 90\%$.

Standby mode power results are shown in Fig. 7 for both the bit-cell only stack, and the method when stacking bit-cells, and powering down the peripheral circuits. In the case when powering down the peripheral circuits, the peripheral header switch is simply set to a large size so that the speed matches the case with no peripheral power down. Fig. 7 shows a savings of between $75 - 87\%$ for the bit-cell only stack, and a savings of between $87 - 90\%$ for the stacking case when powering down the peripheral circuits. It is illuminating to see that powering down the decoders does not have a particularly large overall impact on the total power savings, particularly for large memory sizes.

VI Chip Organization

As this work has shown promise to achieve nice power savings with low overhead, we recently sent out a chip for fabrication containing several of these stacked memory banks. Fig. 8 shows the design layout photo. The chip was designed in a fully-depleted SOI, 180nm technology, and includes three stackable SRAM banks. The on-chip design implements the first method discussed, stacking only the SRAM bit cell arrays during sleep, leaving the decoders powered. Two operating modes exist to allow the stacking of either two or three banks.

The power switches were designed in a tunable fashion, as is shown in Fig. 2. The overall area of one switch is approximately 90 x 65 μm, which includes the tunable V_{dd} and ground power switches, and the transmission gate to connect the SRAM bank cell arrays together. The intended result of this chip is to empirically show how well the stacking concept works, and whether it is practical to stack more than two banks. The hope is to further optimize this design, and fabricate this in a cutting-edge technology node, such as a 45nm process.

VII Conclusions and Future Work

This work presented the idea of stacking SRAM banks to save leakage power during standby mode. Simulations were performed in the 65nm PTM technology, showing an overall power savings of 90% during standby. Noise vs. switch area tradeoffs were presented for the virtual supply rails.

Future work consists of optimizing the power switches for the memory and the peripheral circuits, considering area, noise, and possible de-capacitor insertion. Fabrication results will give conclusive results on enter/exit time for sleep mode, and will help determine how many banks can be stacked during sleep.

Acknowledgment

We would like to thank Ben Calhoun, Wei Huang, Zhenyu Qi, and Jiajing Wang from the University of Virginia, and Jan Rabaey from UC Berkeley for illuminating discussions on this topic. This work was supported in part by a grant from Intel, by an NSF CRI (CNS-0551630) grant, and by the Interconnect Focus Center, one of five research centers funded under the Focus Center Research Program, a Semiconductor Research Corporation and DARPA program.

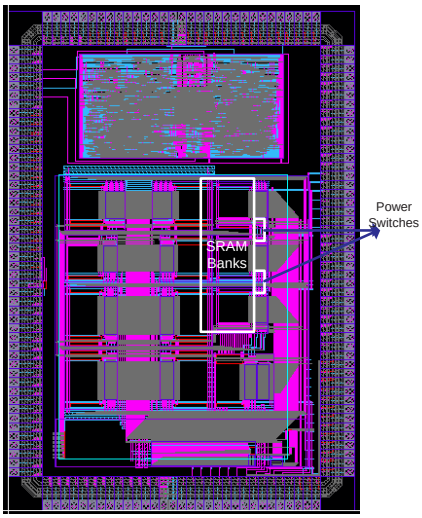

Fig. 8. Finished layout of stacked SRAM test chip sent out for fabrication.

References

[1] N. S. Kim, K. Flautner, D. Blaauw, and T. Mudge, "Drowsy instruction caches," in *Proceedings of 35th IEEE/ACM Intl. Symp. on Microarchitecture*, November 2002, pp. 219–230.

[2] H. Qin, Y. Cao, D. Markovic, A. Vladimirescu, and J. Rabaey, "Standby supply voltage minimization for deep sub-micron sram," *Microelectronics Journal*, pp. 789–800, January 2005.

[3] C. H. Kim, J. Kim, S. Mukhopadhyay, and K. Roy, "A forward body-biased low-leakage sram cache: Device and architecture considerations," in *Proceedings of International Symposium on Low Power Electronics and Design*, August 2003, pp. 6–9.

[4] S. Shigematsu, S. Mutoh, Y. Matsuya, Y. Tanabe, and J. Yamada, "A 1-v high-speed mtcmos circuit scheme for power-down application circuits," *IEEE Journal of Solid State Circuits*, vol. 32, no. 6, pp. 861–869, June 1997.

[5] N. Azizi, F. N. Najm, and A. Moshovos, "Low-leakage asymmetric-cell sram," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 11, no. 4, pp. 701–715, 2003.

[6] B. H. Calhoun and A. P. Chandrakasan, "A 256-kb 65-nm sub-threshold SRAM design for ultra-low-voltage operation," *IEEE Journal of Solid State Circuits*, vol. 42, no. 3, pp. 680–688, March 2007.

[7] S. Rajapandian, K. L. Shepard, P. Hazucha, and T. Karnik, "High-voltage power delivery through charge recycling," *IEEE Journal of Solid State Circuits*, vol. 41, no. 6, pp. 1400–1410, June 2006.

[8] Y. K. Ramadass and A. P. Chandrakasan, "Minimum energy tracking loop with embedded DC-DC converter delivering voltages down to 250mv in 65nm cmos," in *Proc. of IEEE Solid-State Circ. Conf.*, February 2007, pp. 64–65.

[9] M. Seok, S. Hanson, Y.-S. Lin, Z. Foo, D. Kim, L. Lee, N. Liu, D. Sylvester, and D. Blaauw, "The phoenix processor: a 30pw platform for sensor applications," in *Proc. IEEE Symp. on VLSI Tech.*, June 2008, pp. 188–189.

[10] H. Jiang, M. Marek-Sadowska, and S. R. Nassif, "Benefits and costs of power-gating technique," in *Proc. of IEEE Conf. on Comp. Des.*, October 2005, pp. 559–566.

[11] L. Di, M. Putic, J. Lack, and B. H. Calhoun, "Power switch characterization for fine-grained dynamic voltage scaling," in *Proc. of IEEE Conf. on Comp. Des.*, October 2008, pp. 605–611.

[12] E. Seevinck, F. J. List, and J. Lohstroh, "Static-noise margin analysis of mos sram cells," *IEEE Journal of Solid State Circuits*, vol. SC-22, no. 5, pp. 748–754, October 1987.

[13] J. Wang, A. Singhee, R. Rutenbar, and B. H. Calhoun, "Statistical modeling for the minimum standby supply voltage of a full SRAM array," in *Proc. of European Sol. State Circ. Conf.*, September 2007, pp. 400–403.